



RECEIVED

JUN 09 2004

AMENDMENTS TO THE CLAIMS:

Technology Center 2100

IN THE CLAIMS:

1 1. (currently amended) A method for recoverable programming, comprising the steps  
2 of:  
3 identifying a predetermined instruction sequence containing a memory access  
4 request;  
5 checkpointing a predetermined set of system data;  
6 executing the memory access request after the checkpointing;  
7 monitoring for memory access errors;  
8 logging a any memory access ~~error~~ errors in an error logging register;  
9 polling the register for any logged memory access error during execution of the  
10 instruction sequence; and  
11 raising exceptions, if ~~the~~ any memory access error is logged; and  
12 recovering from any memory access error using the checkpointed system data, if  
13 the memory access error is logged during execution of the instruction sequence.

1 2. (canceled) The method of claim 1, further comprising the steps of:  
2 checkpointing a predetermined set of system data; and  
3 recovering from the memory access error using the checkpointed system data, if  
4 the memory access error is logged during execution of the instruction sequence.

1 3. (original) The method of claim 1, further comprising the step of:

06/03/2004 WABDELRI 00000003 082025 09845469

01 FC:1201 258.00 DA  
02 FC:1202 36.00 DA

2            setting data returned in response to the memory access request equal to a set of  
3 predefined fake data, if the memory access error is logged during execution of the  
4 instruction sequence.

1    4.        (original) The method of claim 3, further comprising the step of:  
2            skipping the polling and raising steps if the data returned in response to the  
3 memory access request is not equivalent to the predefined fake data.

1    5.        (original) The method of claim 1, further comprising the step of:  
2            masking a machine check abort handle.

1    6.        (original) The method of claim 5, after the raising step, further comprising the  
2 steps of:  
3            enabling the machine check abort handle.

1    7.        (original) The method of claim 1, further comprising the step of:  
2            updating pointers, if the memory access error is logged.

1    8.        (original) The method of claim 1, further comprising the step of:  
2            re-executing the memory access request, if software so commands.

1    9.        (original) A method for recoverable programming, comprising the steps of:  
2            identifying a predetermined instruction sequence;  
3            checkpointing a predetermined set of system data;

4           masking a machine check abort handle;  
5           monitoring for memory access errors;  
6           logging a memory access error in an error logging register;  
7           polling the register for any logged memory access error during execution of the  
8 instruction sequence;  
9           raising exceptions, if the memory access error is logged;  
10          updating pointers, if the memory access error is logged;  
11          recovering from the memory access error using the checkpointed system data, if  
12 the memory access error is logged during execution of the instruction sequence.;  
13          re-executing the memory access request, if software so commands; and  
14          enabling the machine check abort handle.

1   10.    (currently amended) A computer-usable medium embodying computer program  
2 code for commanding a computer to perform recoverable programming, comprising the  
3 steps of:  
4          identifying a predetermined instruction sequence containing a memory access  
5 request;  
6          checkpointing a predetermined set of system data;  
7          executing the memory access request after the checkpointing;  
8          monitoring for memory access errors;  
9          logging a any memory access ~~error~~ errors in an error logging register;  
10          polling the register for any logged memory access error during execution of the  
11 instruction sequence; ~~and~~  
12          raising exceptions, if ~~the~~ any memory access error is logged; and

13            recovering from any memory access error using the checkpointed system data, if  
14   the memory access error is logged during execution of the instruction sequence.

1   11.    (canceled) The medium of claim 10, further comprising the steps of:  
2            checkpointing a predetermined set of system data; and  
3            recovering from the memory access error using the checkpointed system data, if  
4   the memory access error is logged during execution of the instruction sequence..

1   12.    (original) The medium of claim 10, further comprising the step of:  
2            setting data returned in response to the memory access request equal to a set of  
3   predefined fake data, if the memory access error is logged during execution of the  
4   instruction sequence.

1   13.    (original) The medium of claim 13, further comprising the step of:  
2            skipping the polling and raising steps if the data returned in response to the  
3   memory access request is not equivalent to the predefined fake data.

1   14.    (original) The medium of claim 10, further comprising the step of:  
2            masking a machine check abort handle.

1   15.    (currently amended) A system for recoverable programming, comprising:  
2            means for identifying a predetermined instruction sequence containing a memory  
3   access request;  
4            means for checkpointing a predetermined set of system data;

5        means for executing the memory access request after the checkpointing;  
6        means for monitoring for memory access errors;  
7        means for logging ~~a~~ any memory access ~~error~~ errors in an error logging register;  
8        means for polling the register for any logged memory access error during  
9        execution of the instruction sequence; ~~and~~  
10       means for raising exceptions, if ~~the~~ any memory access error is logged; and  
11       means for recovering from any memory access error using the checkpointed  
12       system data, if the memory access error is logged during execution of the instruction  
13       sequence.

1    16.    (canceled) The system of claim 15, further comprising:  
2        means for checkpointing a predetermined set of system data; and  
3        means for recovering from the memory access error using the checkpointed  
4        system data, if the memory access error is logged during execution of the instruction  
5        sequence..

1    17.    (original) The system of claim 15, further comprising:  
2        means for setting data returned in response to the memory access request equal to  
3        a set of predefined fake data, if the memory access error is logged during execution of the  
4        instruction sequence.

1    18.    (original) The system of claim 17, further comprising:

2 means for bypassing the means for polling and means for raising if the data  
3 returned in response to the memory access request is not equivalent to the predefined fake  
4 data.

1 19. (original) The system of claim 15, further comprising the step of:

2 means for masking a machine check abort handle.

1 20. (new) A method for recoverable programming, comprising the steps of:

2 identifying a predetermined instruction sequence;

3 monitoring for memory access errors;

4 logging a memory access error in an error logging register;

5 polling the register for any logged memory access error during execution of the

6 instruction sequence;

7 raising exceptions, if the memory access error is logged; and

8 setting data returned in response to the memory access request equal to a set of

9 predefined fake data, if the memory access error is logged during execution of the

10 instruction sequence.

1 21. (new) The method of claim 20, further comprising the step of:

2 skipping the polling and raising steps if the data returned in response to the

3 memory access request is not equivalent to the predefined fake data.

1 22. (new) A computer-usable medium embodying computer program code for

2 commanding a computer to perform recoverable programming, comprising the steps of:

3 identifying a predetermined instruction sequence;  
4 monitoring for memory access errors;  
5 logging a memory access error in an error logging register;  
6 polling the register for any logged memory access error during execution of the  
7 instruction sequence;  
8 raising exceptions, if the memory access error is logged; and  
9 setting data returned in response to the memory access request equal to a set of  
10 predefined fake data, if the memory access error is logged during execution of the  
11 instruction sequence.

1 23. (new) The medium of claim 22, further comprising the step of:  
2 skipping the polling and raising steps if the data returned in response to the  
3 memory access request is not equivalent to the predefined fake data.

1 24. (new) A system for recoverable programming, comprising:  
2 means for identifying a predetermined instruction sequence;  
3 means for monitoring for memory access errors;  
4 means for logging a memory access error in an error logging register;  
5 means for polling the register for any logged memory access error during  
6 execution of the instruction sequence;  
7 means for raising exceptions, if the memory access error is logged; and  
8 means for setting data returned in response to the memory access request equal to  
9 a set of predefined fake data, if the memory access error is logged during execution of the  
10 instruction sequence.

- 1 25. (new) The system of claim 24, further comprising:
- 2 means for bypassing the means for polling and means for raising if the data
- 3 returned in response to the memory access request is not equivalent to the predefined fake
- 4 data.